ENV 790.30 - Time Series Analysis for Energy Data | Spring 2023

Assignment 5 - Due date 02/27/23

Yuxiang Ren

Directions

You should open the .rmd file corresponding to this assignment on RStudio. The file is available on our class repository on Github. And to do so you will need to fork our repository and link it to your RStudio.

Once you have the file open on your local machine the first thing you will do is rename the file such that it includes your first and last name (e.g., "LuanaLima_TSA_A05_Sp23.Rmd"). Then change "Student Name" on line 4 with your name.

Then you will start working through the assignment by **creating code and output** that answer each question. Be sure to use this assignment document. Your report should contain the answer to each question and any plots/tables you obtained (when applicable).

When you have completed the assignment, **Knit** the text and code into a single PDF file. Submit this pdf using Sakai.

R packages needed for this assignment: "xlsx" or "readxl", "ggplot2", "forecast","tseries", and "Kendall". Install these packages, if you haven't done yet. Do not forget to load them before running your script, since they are NOT default packages.\

```
#Load/install required package here
library(xlsx)
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
library(tseries)
library(ggplot2)
library(Kendall)
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(tidyverse)  #load this package so yon clean the data frame using pipes
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2
## --

## v tibble  3.1.8     v dplyr   1.1.0
## v tidyr   1.3.0     v stringr 1.5.0
## v readr   2.1.4     v forcats 1.0.0
## v purrr   1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x lubridate::as.difftime() masks base::as.difftime()
## x lubridate::date()        masks base::date()
## x dplyr::filter()          masks stats::filter()
## x lubridate::intersect()   masks base::intersect()
## x dplyr::lag()             masks stats::lag()
## x lubridate::setdiff()     masks base::setdiff()
## x lubridate::union()       masks base::union()
```

Decomposing Time Series

Consider the same data you used for A04 from the spreadsheet "Table_10.1_Renewable_Energy_Production_and_Consump

```r
#Importing data set - using xlsx package
energy_data <- read.xlsx(file=
                           "./Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source.xls

#Now let's extract the column names from row 11 only
read_col_names <- read.xlsx(file=
                              "./Data/Table_10.1_Renewable_Energy_Production_and_Consumption_by_Source.

colnames(energy_data) <- read_col_names
head(energy_data)
```

```
##        Month Wood Energy Production Biofuels Production
## 1 1973-01-01                129.630       Not Available
## 2 1973-02-01                117.194       Not Available
## 3 1973-03-01                129.763       Not Available
## 4 1973-04-01                125.462       Not Available
## 5 1973-05-01                129.624       Not Available
## 6 1973-06-01                125.435       Not Available
##   Total Biomass Energy Production Total Renewable Energy Production
## 1                         129.787                           403.981
## 2                         117.338                           360.900
## 3                         129.938                           400.161
## 4                         125.636                           380.470
## 5                         129.834                           392.141
## 6                         125.611                           377.232
##   Hydroelectric Power Consumption Geothermal Energy Consumption
## 1                         272.703                         1.491
## 2                         242.199                         1.363
## 3                         268.810                         1.412
## 4                         253.185                         1.649
## 5                         260.770                         1.537
## 6                         249.859                         1.763
```

```
##   Solar Energy Consumption Wind Energy Consumption Wood Energy Consumption
## 1            Not Available           Not Available                 129.630
## 2            Not Available           Not Available                 117.194
## 3            Not Available           Not Available                 129.763
## 4            Not Available           Not Available                 125.462
## 5            Not Available           Not Available                 129.624
## 6            Not Available           Not Available                 125.435
##   Waste Energy Consumption Biofuels Consumption
## 1                    0.157        Not Available
## 2                    0.144        Not Available
## 3                    0.176        Not Available
## 4                    0.174        Not Available
## 5                    0.210        Not Available
## 6                    0.176        Not Available
##   Total Biomass Energy Consumption Total Renewable Energy Consumption
## 1                          129.787                            403.981
## 2                          117.338                            360.900
## 3                          129.938                            400.161
## 4                          125.636                            380.470
## 5                          129.834                            392.141
## 6                          125.611                            377.232
```

```
nobs=nrow(energy_data)
nvar=ncol(energy_data)
```

Q1

For this assignment you will work only with the following columns: Solar Energy Consumption and Wind Energy Consumption. Create a data frame structure with these two time series only and the Date column. Drop the rows with *Not Available* and convert the columns to numeric. You can use filtering to eliminate the initial rows or convert to numeric and then use the drop_na() function. If you are familiar with pipes for data wrangling, try using it!

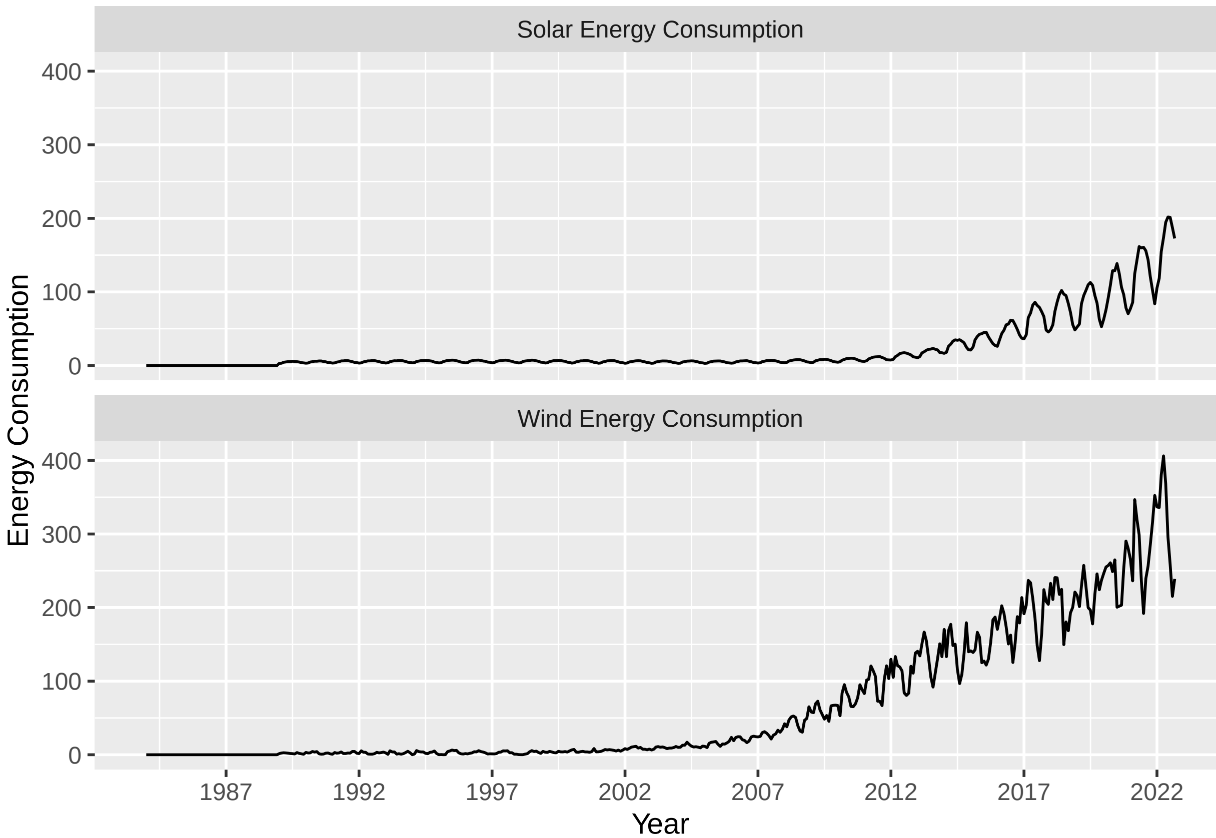
```
A05_raw <- energy_data %>%
  select(Month, 'Solar Energy Consumption', 'Wind Energy Consumption') %>%
  filter(`Solar Energy Consumption` != "Not Available" &
           `Wind Energy Consumption` != "Not Available")

A05_raw$`Solar Energy Consumption` <- as.numeric(A05_raw$`Solar Energy Consumption`)
A05_raw$`Wind Energy Consumption` <- as.numeric(A05_raw$`Wind Energy Consumption`)
```

Q2

Plot the Solar and Wind energy consumption over time using ggplot. Plot each series on a separate graph. No need to add legend. Add informative names to the y axis using `ylab()`. Explore the function scale_x_date() on ggplot and see if you can change the x axis to improve your plot. Hint: use *scale_x_date(date_breaks = "5 years", date_labels = "%Y")")*

```
# pivot_longer
A05_raw_longer <- pivot_longer(A05_raw,'Solar Energy Consumption':'Wind Energy Consumption',
                               names_to = "Energy", values_to = "Consumption")
```

```r
Plot_Q2 <- ggplot(A05_raw_longer, aes(Month, Consumption))+
  geom_line()+
  ylab('Energy Consumption')+
  xlab('Year')+
  scale_x_date(date_breaks = "5 years", date_labels = "%Y") +
  facet_wrap(vars(Energy), nrow = 2)
print(Plot_Q2)
```



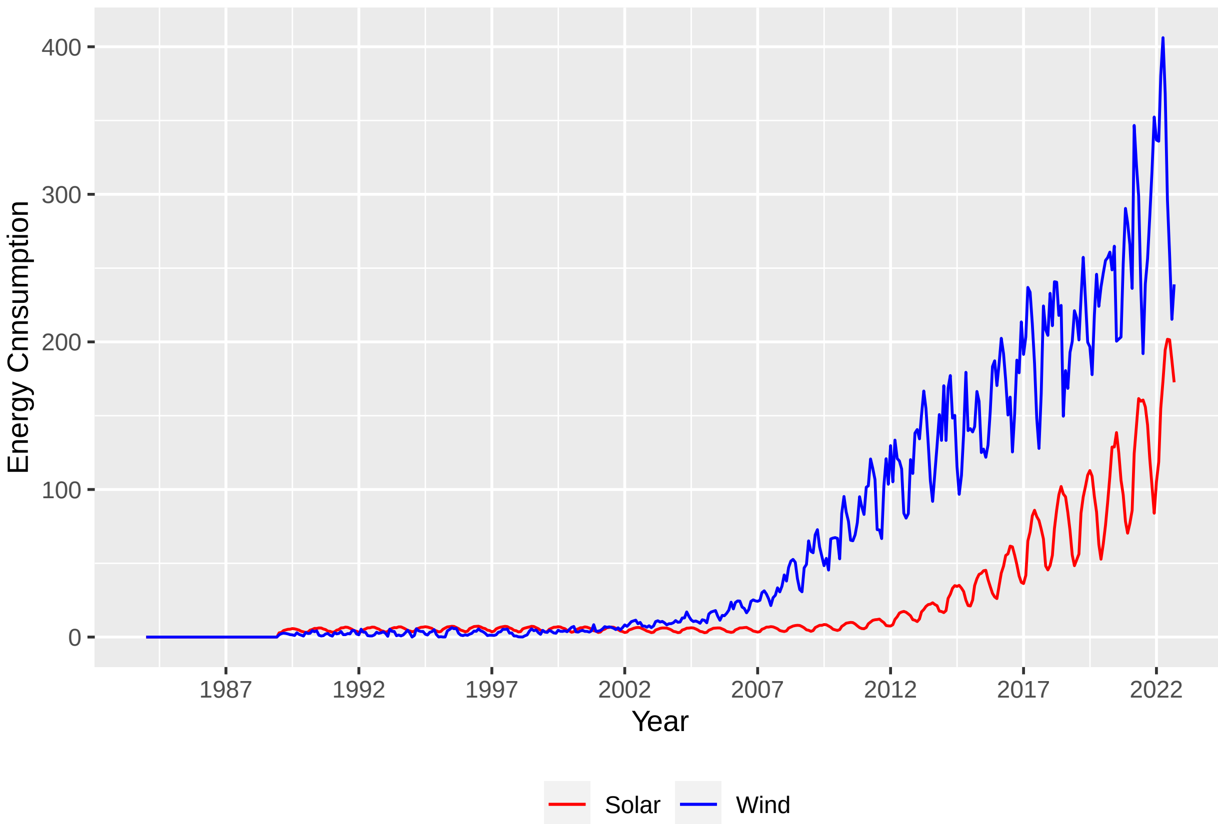
Q3

Now plot both series in the same graph, also using ggplot(). Look at lines 141-148 of the file `M4_OutliersMissingData_Part2_Complete.Rmd` to learn how to manually add a legend to ggplot. Make the solar energy consumption red and wind energy consumption blue. Add informative name to the y axis using `ylab("Energy Consumption)`. And use function scale_x_date() again to improve x axis.

```r
Plot_Q3 <-
ggplot(A05_raw, aes(Month, A05_raw$'Solar Energy Consumption', color = "Solar")) +
  geom_line()+
  geom_line(aes(Month,A05_raw$'Wind Energy Consumption', color = "Wind")) +
  labs(color="") +
  scale_color_manual(values = c("Solar" = "red", "Wind" = "blue"),
                                labels=c("Solar", "Wind")) +
  theme(legend.position = "bottom") +
  ylab(label="Energy Cnnsumption") +
  xlab('Year')+
```

```
  scale_x_date(date_breaks = "5 years", date_labels = "%Y")
print(Plot_Q3)
```

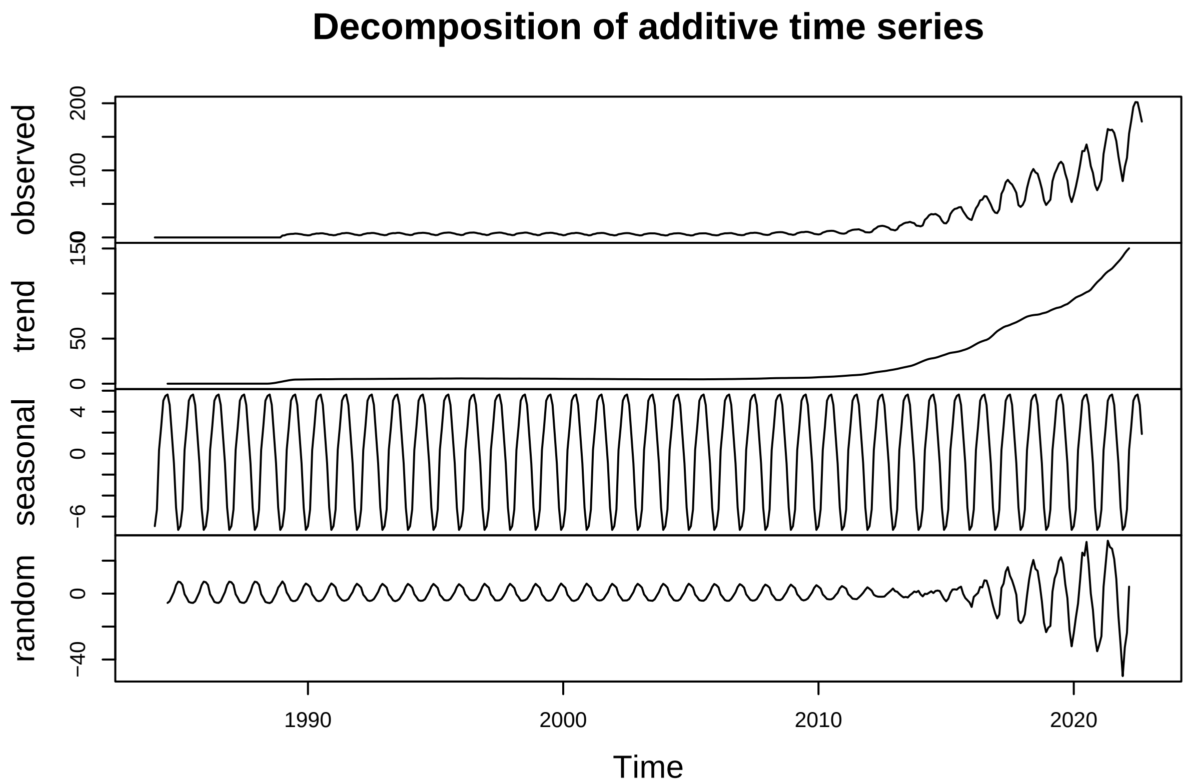
```
## Warning: Use of `A05_raw$"Solar Energy Consumption"` is discouraged.
## i Use `Solar Energy Consumption` instead.

## Warning: Use of `A05_raw$"Wind Energy Consumption"` is discouraged.
## i Use `Wind Energy Consumption` instead.
```



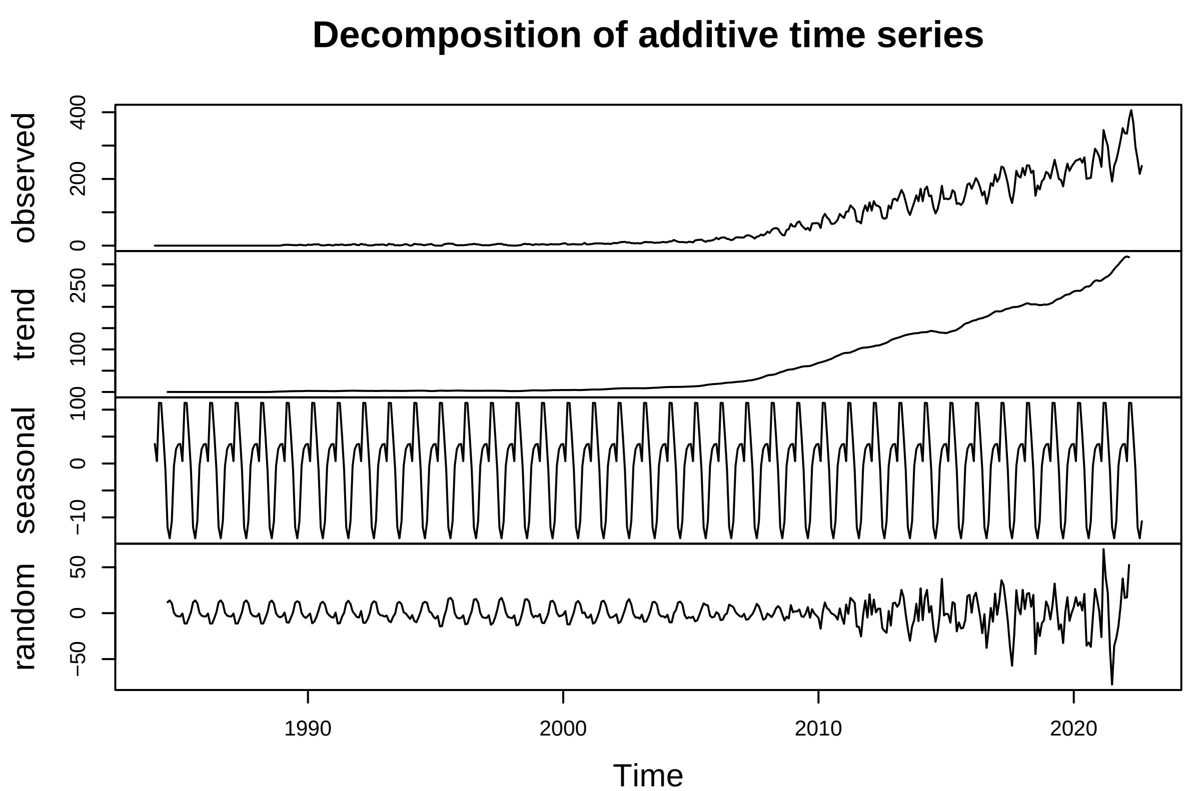
Q4

Transform wind and solar series into a time series object and apply the decompose function on them using the additive option, i.e., `decompose(ts_data, type = "additive")`. What can you say about the trend component? What about the random component? Does the random component look random? Or does it appear to still have some seasonality on it?

```
solar_ts <- ts(A05_raw$`Solar Energy Consumption`, start = c(1984, 1), frequency = 12)
wind_ts <- ts(A05_raw$`Wind Energy Consumption`, start = c(1984, 1), frequency = 12)
solar_decomp <- decompose(solar_ts, type = "additive")
wind_decomp <- decompose(wind_ts, type = "additive")
plot(solar_decomp)
```




```
plot(wind_decomp)
```



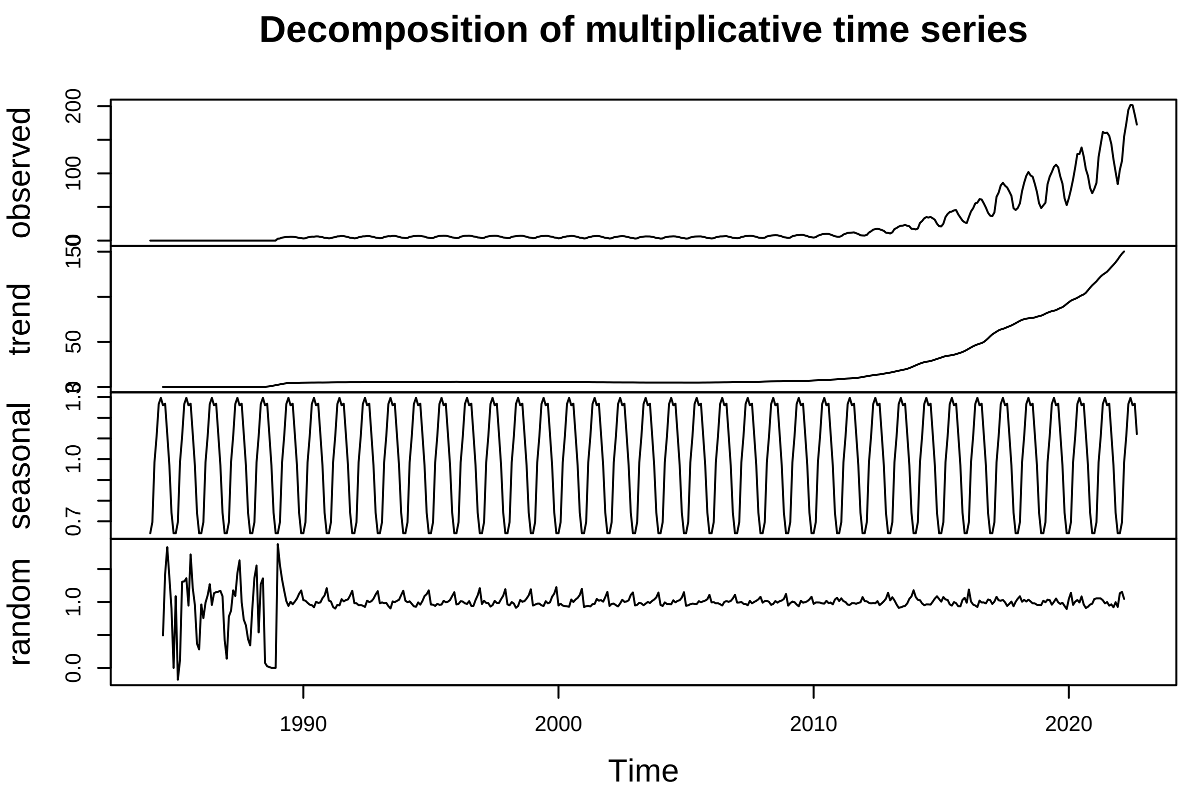
>

Answer: Trends in both solar and wind are relatively simple, roughly linear, increasing trends. However, the random component does not look random. There is obvious seasonality in it, especially in solar energy consumption. Additionally, in solar energy, this seasonal fluctuation in random components was small and stable before 2015, while after 2015, the fluctuation became larger year by year. Moreover, the intensification of seasonal fluctuations in random components began in 2010 in wind.

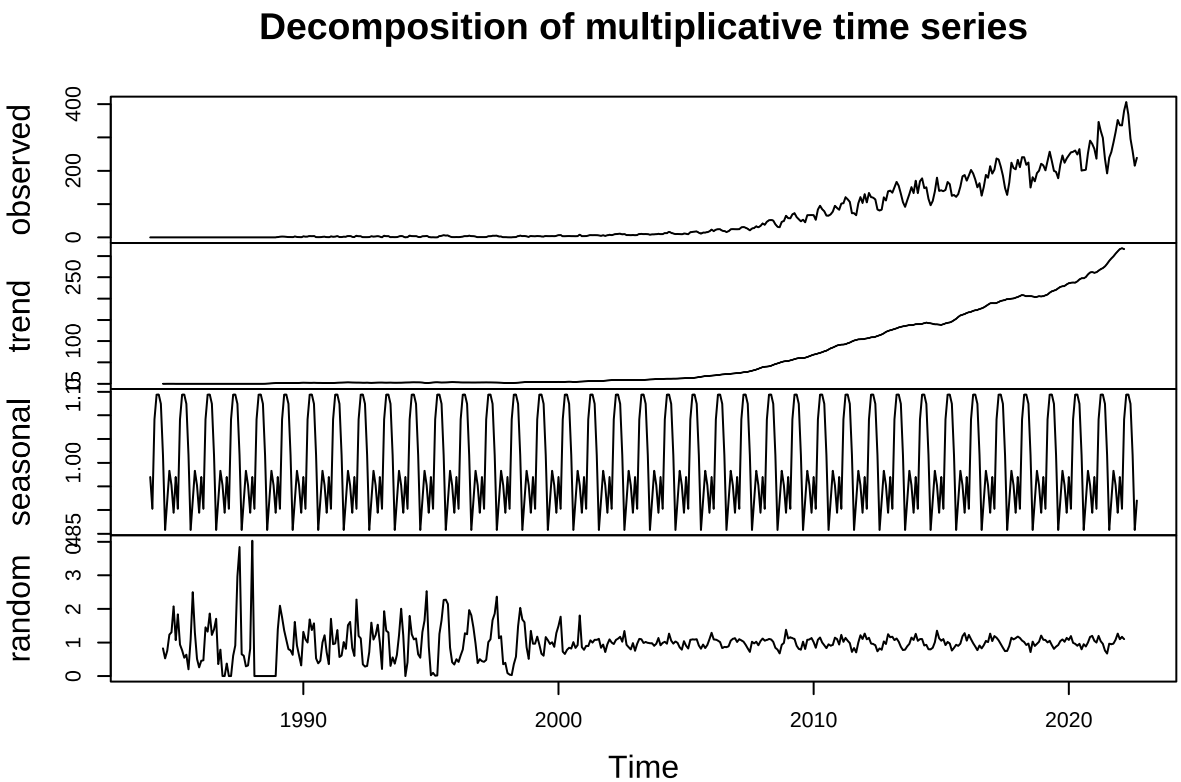
Q5

Use the decompose function again but now change the type of the seasonal component from additive to multiplicative. What happened to the random component this time?

```
solar_decomp_mult <- decompose(solar_ts, type = "multiplicative")
wind_decomp_mult <- decompose(wind_ts, type = "multiplicative")
plot(solar_decomp_mult)
```




```
plot(wind_decomp_mult)
```



>

Answer: All values in random become greater than 0. For both energy sources, the range of large fluctuations is no longer the recent years but the earliest years. For example, the large-scale change point of solar energy appeared before 1990, and the emergence time of wind energy was before 2000. The fluctuation of the values in the adjacent years has become smaller, and the seasonality is not as prominent as before.

Q6

When fitting a model to this data, do you think you need all the historical data? Think about the data from 90s and early 20s. Are there any information from those years we might need to forecast the next six months of Solar and/or Wind consumption. Explain your response.

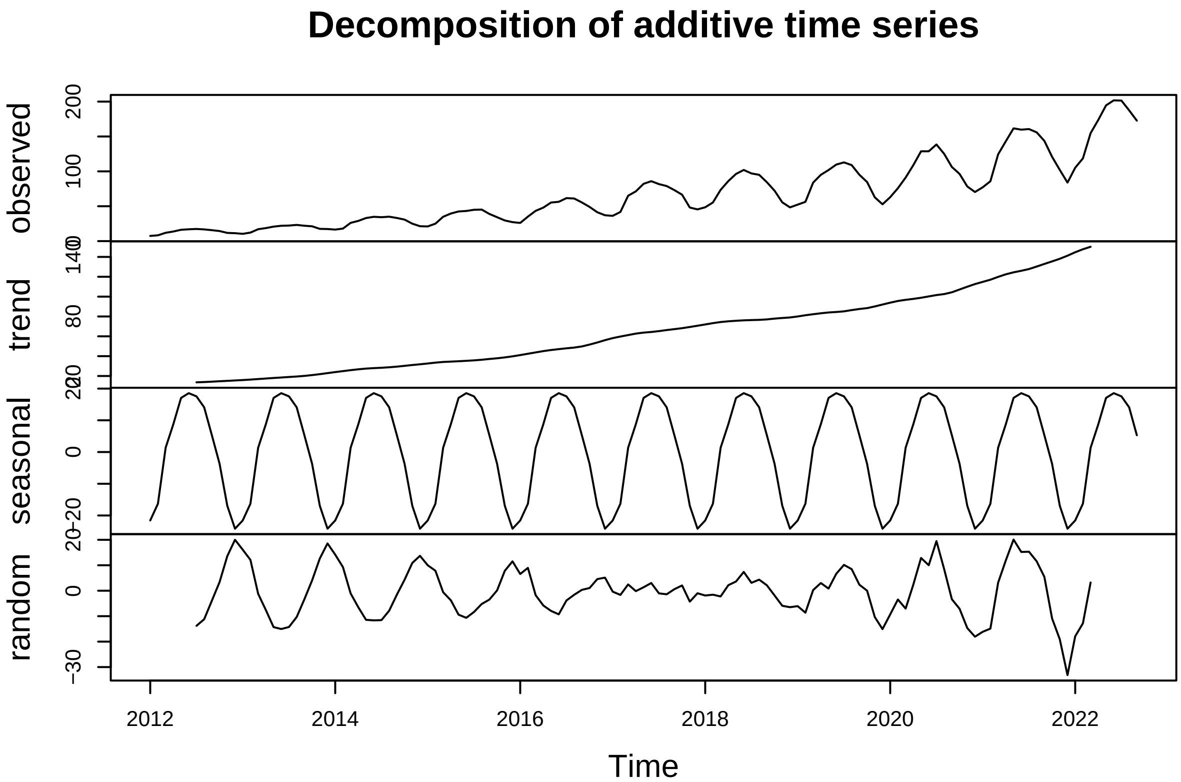
> Answer: I don't think there need all the historical data. Because the data values for the earlier years are small and have a small trend value, their influence on the forecast for the next 6 months is small and might be ignored.

Q7

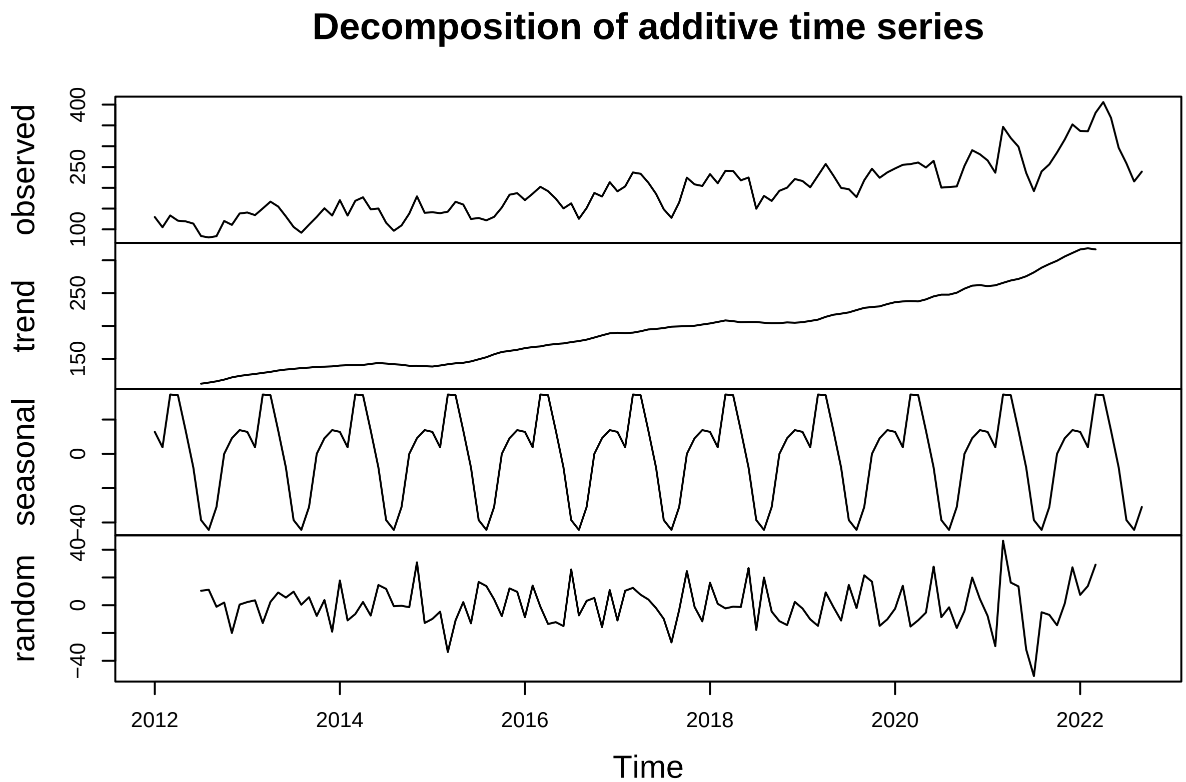
Create a new time series object where historical data starts on January 2012. Hint: use `filter()` function so that you don't need to point to row numbers, .i.e, `filter(xxxx, year(Date) >= 2012 )`. Apply the decompose function `type=additive` to this new time series. Comment the results. Does the random component look random? Think about our discussion in class about seasonal components that depends on the level of the series.

```
A05_2012 <- filter(A05_raw, year(Month) >= 2012 )
solar_ts_2012 <- ts(A05_2012$`Solar Energy Consumption`, start = c(2012, 1), frequency = 12)
wind_ts_2012 <- ts(A05_2012$`Wind Energy Consumption`, start = c(2012, 1), frequency = 12)
solar_decomp_2012 <- decompose(solar_ts_2012, type = "additive")
```

```
wind_decomp_2012 <- decompose(wind_ts_2012, type = "additive")
plot(solar_decomp_2012)
```




```
plot(wind_decomp_2012)
```



Answer: The random component of the two energy sources no longer looks like it used to have two distinct phases, a stable phase and a huge fluctuation phase. For solar energy, there is still some seasonal fluctuation in the random part. While, such seasonal fluctuations are not as pronounced in wind energy in random section.